AF

PATENT

DOCKET NO.: 05-03-014
CLIENT NO.: UGSC01-05031
Customer No.: 45113

OIPE IAP85
JUL 3 0 2007
PATENT & TRADEMARK OFFICE

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of	:	Jianbing Huang, et al.
U.S. Serial No.	:	10/731,281
Filed	:	December 9, 2003
For	:	SYSTEM AND METHOD FOR TRANSPARENCY RENDERING
Group No.	:	2628
Examiner	:	Kimbinh T. Nguyen

MAIL STOP APPEAL BRIEF - PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

CERTIFICATE OF MAILING BY FIRST CLASS MAIL

The undersigned hereby certifies that the following documents:

1. Reply Brief; and
2. Postcard receipt

relating to the above application, were deposited as "First Class Mail" with the United States Postal Service, addressed to MAIL STOP APPEAL BRIEF - PATENTS, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on July 25, 2007.

Date: 7-25-07 Elizabeth Ahern
Mailer

Date: 7/25/7 Matthew S. Anderson
Matthew S. Anderson
Reg. No. 39,093

P.O. Drawer 800889
Dallas, Texas 75380
Phone: (972) 628-3600
Fax: (972) 628-3616
E-mail: *manderson@munckbutrus.com*

OIPE IAP85 JUL 30 2007 PATENT & TRADEMARK OFFICE

DOCKET NO.: 05-03-014
CLIENT NO.: UGSC01-05031
Customer No.: 45113

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	Jianbing Huang, et al.
Serial No.:	10/731,281
Filed:	December 9, 2003
For:	SYSTEM AND METHOD FOR TRANSPARENCY RENDERING
Group No.:	2628
Examiner:	Kimbinh T. Nguyen

MAIL STOP APPEAL BRIEF - PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

REPLY BRIEF

Sir:

This Reply Brief is filed in response to the Examiner's Answer mailed May 25, 2007, to address arguments made in that Answer. This Reply Brief does not include any new or non-admitted amendment, nor any new or non-admitted affidavit or other evidence. No fees are believed due, but please charge any additional necessary fees to Deposit Account No. 50-0208.

TABLE OF CONTENTS

TABLE OF AUTHORITIES

Real Party in Interest

The real party in interest, and assignee of this case, is UGS Corp.

Related Appeals or Interferences

To the best knowledge and belief of the undersigned attorney, there are none.

Status of Claims

Claims 1-30 are under final rejection, and are each appealed. Applicant notes that while the Office Action Summary sent with the final Office Action only indicates that claims 1-24 were rejected, a rejection was made to all claims.

Status of Amendments after Final

An after-final Amendment to correct a typographic error in Claim 24 was entered, and is reflected in the claims appendix.

SUMMARY OF CLAIMED SUBJECT MATTER

The following summary refers to disclosed embodiments and their advantages, but does not delimit any of the claimed inventions.

The description below is identical to that found the in the Appeal Brief, and is included simply to fulfill the formal requirements of a brief on appeal.

In General

The present application is directed, in general, to graphics processing, and, more specifically, to a system, method, and computer program product that accepts raw polygon geometry and view parameters from a visualization API, sorts the polygons in back-to-front order, and then supplies the sorted triangles to graphics API. *Page 1, lines 4-5, and page 4, lines 2-6.*

Support for Independent Claims

Note that, per 37 CFR §41.37, only each of the independent claims and claims including means-plus-function language are discussed in this section. In the arguments below, however, the dependent claims are also discussed and distinguished from the prior art. The discussion of the claims is for illustrative purposes, and is not intended to effect the scope of the claims.

Independent Claim 1 describes method for graphics processing, including receiving node and view data for a graphic object (1005). The method also includes building a binary-space-partition tree corresponding to the graphic object (1010), the binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf. The method also includes sorting shapes at each leaf of the binary-space-partition tree (1015), and outputting the sorted shapes (1020). *Page 11, lines 18-23; page 32, line 22 - page 33, line 5; and Figure 10.*

Independent Claim 8 describes a method for graphics processing, including analyzing shapes in a graphic object (1105). The method also includes creating a root node and a list of additional nodes for a binary-space-partition tree (1110), each node associated with up to a predetermined number of at least one shape. The method also includes performing a partition plane selection for

each additional node (1115). The method also includes classifying the shapes at the additional node according to the partition plane selection (1120). The method also includes creating child nodes according to the shape classification (1125). *Page 11, lines 18-23; page 33, lines 6 - 23; and Figure 11.*

Independent Claim 11 describes a data processing system having at least a processor (102) and accessible memory (104, 108), and means for performing steps corresponding to the method of claim 1. These include means (122) for receiving node and view data for a graphic object (1005). The system also includes means (102) for building a binary-space-partition tree corresponding to the graphic object (1010), the binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf . The system also includes means (102) for sorting shapes at each leaf of the binary-space-partition tree (1015), and means (122, 110) for outputting the sorted shapes (1020). Dependent Claim 13 further describes means (104, 108) for caching the shape data, and dependent Claim 14 further describes means (102) for traversing the binary-space-partition tree. *Page 8, lines 25 - page 11, line 4; page 11, lines 18-23; page 15, lines 6-18; page 30, line 34 - page 32, line 21; page 32, line 22 - page 33, line 5; and Figures 1 and 10.*

Independent Claim 18 describes a data processing system having at least a processor (102) and accessible memory (104, 108), and means for performing steps corresponding to the method of claim 18. These include means (102) for analyzing shapes in a graphic object. The system also includes means (102) for creating a root node and a list of additional nodes for a binary-space-partition tree (1110), each node associated with up to a predetermined number of at least one shape. The system also includes means (102) for performing a partition plane selection for each additional node (1115). The method also includes means for classifying the shapes at the additional node according to the partition plane selection (1120). The method also includes means (102) for creating child nodes according to the shape classification (1125). *Page 8, lines 25 - page 11, line 4; page 11, lines 18-23; page 33, lines 6 - 23; and Figures 1 and 11.*

Independent Claim 21 describes a computer program product tangibly embodied in a machine-readable medium, comprising instructions for performing steps corresponding to the method of claim 1. These include instructions for receiving node and view data for a graphic object (1005) and instructions for building a binary-space-partition tree corresponding to the graphic object

(1010), the binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf. The computer program product further includes instructions for sorting shapes at each leaf of the binary-space-partition tree (1015), and instructions for outputting the sorted shapes (1020). *Page 11, lines 18-23; page 32, line 22 - page 33, line 5; page 35, lines 1-17; and Figure 10.*

Independent Claim 28 describes a computer program product tangibly embodied in a machine-readable medium, comprising instructions for performing steps corresponding to the method of claim 8. These include instructions for creating a root node and a list of additional nodes for a binary-space-partition tree (1110), each node associated with up to a predetermined number of at least one shape. The computer program product further includes instructions for performing a partition plane selection for each additional node (1115). The computer program product further includes instructions for classifying the shapes at the additional node according to the partition plane selection (1120). The computer program product further includes instructions for creating child nodes according to the shape classification (1125). *Page 11, lines 18-23; page 33, lines 6 - 23; page 35, lines 1-17; and Figure 11.*

Grounds of Rejection to be Reviewed on Appeal

1. Are Claims 1, 2, 4, 5, 8-12, 14, 15, 18-22, 24, 25, and 28-30 obvious over U.S. Patent No. 6,912,293 to Korobkin ("Korobkin") in view of U.S. Patent No. 6,624,810 to Brokenshire *et al.* ("Brokenshire")?

2. Are Claims 3, 6, 7, 13, 16, 17, 23, 26, and 27 obvious over U.S. Patent No. 6,912,293 to Korobkin ("Korobkin") in view of U.S. Patent No. 6,624,810 to Brokenshire *et al.* ("Brokenshire") and further in view of U.S. Patent Application Publication No. 2004/0114794 to Vlasic *et al.* ("Vlasic")?

ARGUMENT

Stated Grounds of Rejection

The rejections outstanding against the Claims are as follows:

Claims 1, 2, 4, 5, 8-12, 14, 15, 18-22, 24, 25, and 28-30 were rejected as obvious over U.S. Patent No. 6,912,293 to Korobkin ("Korobkin") in view of U.S. Patent No. 6,624,810 to Brokenshire et al. ("Brokenshire").

Claims 3, 6, 7, 13, 16, 17, 23, 26, and 27 were rejected as obvious over U.S. Patent No. 6,912,293 to Korobkin ("Korobkin") in view of U.S. Patent No. 6,624,810 to Brokenshire et al. ("Brokenshire") and further in view of U.S. Patent Application Publication No. 2004/0114794 to Vlasic et al. ("Vlasic").

Each claim under each ground of rejection is addressed separately.

Legal Standards

In rejecting claims under 35 U.S.C. § 103(a), the examiner bears the initial burden of establishing a *prima facie* case of obviousness. (*In re Oetiker*, 977 F.2d 1443, 1445, 24 USPQ2d 1443, 1444 (Fed. Cir. 1992), See also *In re Piasecki*, 745 F.2d 1468, 1472, 223 USPQ 785, 788 (Fed. Cir. 1984)). It is incumbent upon the examiner to establish a factual basis to support the legal conclusion of obviousness. (*Id.* at 1073, 5 USPQ2d at 1598). In so doing, the examiner is expected to make the factual determinations set forth in *Graham v. John Deere Co.*, 383 U.S. 1, 17, 148 USPQ 459, 467 (1966), *viz*., (1) the scope and content of the prior art; (2) the differences between the prior art and the claims at issue; and (3) the level of ordinary skill in the art. In addition to these factual determinations, the examiner must also provide "some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness." (*In re Kahn*, 441 F.3d 977, 988, 78 USPQ2d 1329, 1336 (Fed. Cir 2006) (cited with approval in *KSR Int'l v. Teleflex Inc.*, 127 S. Ct. 1727, 1741, 82 USPQ2d 1385, 1396 (2007).

Evidence of a motivation to combine prior art references must be clear and particular if the trap of "hindsight" is to be avoided. *In re Dembiczak*, 175 F.3d 994, 50 USPQ2d 1614 (Fed.Cir. 1999) (Evidence of a suggestion, teaching or motivation to combine prior art references must be "clear and particular." "Broad conclusory statements regarding the teaching of multiple references, standing alone, are not 'evidence.'"). *In re Roufett*, 149 F.3d 1350, 1357, 47 USPQ2d 1453, 1457 (Fed.Cir. 1998) ("[R]ejecting patents solely by finding prior art corollaries for the claimed elements would permit an examiner to use the claimed invention itself as a blueprint for piecing together

elements in the prior art to defeat the patentability of the claimed invention. Such an approach would be 'an illogical and inappropriate process by which to determine patentability.'")

Analysis of Examiner's Rejection

The arguments and discussion below are limited to addressing issues raised in the Examiner's Answer. The arguments and discussion included in the Appeal Brief are hereby incorporated by reference.

Ground of Rejection 1: Claims 1, 2, 4, 5, 8-12, 14, 15, 18-22, 24, 25, and 28-30 were rejected as obvious over U.S. Patent No. 6,912,293 to Korobkin ("Korobkin") in view of U.S. Patent No. 6,624,810 to Brokenshire et al. ("Brokenshire").

Claim 1

Claim 1 requires, among other limitations, "building a binary-space-partition tree corresponding to the graphic object, the binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf."

Examiner Nguyen has repeatedly conceded, in examination of this application, that Korobkin fails to teach this limitation as required by claim 1. In the Examiner's Answer, Examiner Nguyen for the first time includes a statement, with absolutely no support whatsoever, that

> Korobkin implicitly discloses that U-regions as leafs. Insertion of geometry stops; it means the partition having up to a predetermined number and stops the partition; as one of ordinary skill in the art, the partition of the binary space tree should be continued until a desired

> number of levels of subspace or sub-object has been created and stopped the partition..." *Examiner's Answer, pages 3-4.*

To the extent this statement is understood, it is incorrect. Korobkin teaches that screen of the "camera" is partitioned into regions that are occupied with projected geometry (G-regions), and those unoccupied (U-regions). The G-regions and U-regions can appear as leaf nodes in this tree. Korobkin teaches that insertion of geometry into the screen stops when there are no unoccupied areas (U-regions) left, which occurs when the screen is full. *See Korobkin, col. 15, line 54 – col. 16, line 3.*

Korobkin has no teaching at all related to sub-space, or levels of subspace, contrary to the Examiner's statement of what is "implied."

Moreover, Claim 1 requires, in addition to the binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf, the step of "sorting shapes at each leaf of the binary-space-partition tree". This is not taught or suggested by Korobkin or Brokenshire.

For this teaching, Examiner Nguyen cites to Korobkin at col, 15, lines 33-36. This passage teaches

> The system utilizes a 3D binary space partition (BSP) tree to accomplish a global visibility sort of the input database. The BSP tree sort recursively subdivides the object space and geometry with hyper-planes defined by the surface facets of the input From any given arbitrary viewpoint, a subsequent traversal of the tree will deliver triangles in a spatially correct "back-to-front" or

> "front-to-back" ordering. *Korobkin, col. 15, lines 32-38.*

Note that there is no explicit sorting taught here; the teaching is that a traversal of the tree delivers triangles in a "spatially correct" ordering. While this may be regarded as a type of "sorting", this is a sorting <u>of the leaves of the tree</u>, not of the <u>shapes at each leaf of the tree</u>, as required by the claim.

This difference is highlighted by the specification as filed, which teaches:

> Specifically, in at least some embodiments, a BSP tree is constructed with multiple-triangle leaf nodes (as opposed to a BSP tree with single-triangle leaf nodes as in conventional practices), and the triangles are sorted by traversing the BSP tree and depth-sorting triangles on each leaf node while it is being traversed. *Specification, page 11, lines 12-17*

Korobkin teaches a BSP tree with single-triangle leaf nodes, precisely as described as "conventional" in the specification, and simply traverses the tree for a "spatially correct" ordering. Korobkin does not have multiple-triangle leaf nodes, and does not sort shapes <u>at each leaf of the binary-space-partition tree</u>, as required by the claim.

Brokenshire similar fails to teach or disclose sorting shapes at each leaf of the binary-space-partition tree, as required by the claim. As such, even the proposed combination of Korobkin and Brokenshire not teach or suggest this claim limitation.

Moreover, there is no suggestion or motivation for one skilled in the art, having common sense and common creativity, to combine discrete elements from Korobkin and then <u>seek out</u> still other teachings not included in these references as required by Claims.

It is clear, then, that no combination of Korobkin, Brokenshire, and Vlasic (or any other cited art), can meet the limitations of this claim. This claim is simply not taught or suggested by any art of record, alone or in combination, and there is no motivation to combine these references to produce a feature not found in any of them. As such, the rejection should be reversed, and this claim and all its dependent claims should be allowed over the art of record.

Claim 8

Independent Claim 8 requires:

A method for graphics processing, comprising:

analyzing shapes in a graphic object;

creating a root node and a list of additional nodes for a binary-space-partition tree, each node associated with up to a predetermined number of at least one shape;

performing a partition plane selection for each additional node,

classifying the shapes at the additional node according to the partition plane selection; and

creating child nodes according to the shape classification.

This claim requires, among other limitations, classifying the shapes at the additional node according to the partition plane selection. The Examiner simply refers to Korobkins' teaching regarding the G-regions and U-regions in the camera view, where each region that is occupied with projected geometry is designated a G-region, and unoccupied regions are U-regions. Even if this were regarded as a "partition plane selection" for the U-region and G-region nodes, no shapes at the nodes are classified according to this partition plane selection, as claimed, and no child nodes are

created according to any such shape classification, as claimed.

It is clear, then, that no combination of Korobkin, Brokenshire, and Vlasic (or any other cited art), can meet the limitations of this claim. This claim is simply not taught or suggested by any art of record, alone or in combination. As such, the rejection should be reversed, and this claim and all its dependent claims should be allowed over the art of record.

Claim 11

Claim 11 requires:

A data processing system having at least a processor and accessible memory, comprising:

means for receiving node and view data for a graphic object;

means for building a binary-space-partition tree corresponding to the graphic object, the binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf;

means for sorting shapes at each leaf of the binary-space-partition tree; and

means for outputting the sorted shapes.

Examiner Nguyen has repeatedly conceded, in examination of this application, that Korobkin fails to teach a binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf, as required by claim 1. In the Examiner's Answer, Examiner Nguyen for the first time includes a statement, with absolutely no support whatsoever, that this is an implicit teaching of Korobkin, as discussed above with regard to claim 1. The argument on this point of claim 1 is incorporated herein.

Moreover, Claim 11 also requires means for "sorting shapes at each leaf of the binary-space-partition tree". This is not taught or suggested by Korobkin or Brokenshire.

For this teaching, Examiner Nguyen cites to Korobkin at col, 15, lines 33-36, reproduced above with relation to claim 1. There is no explicit sorting taught in this passage; the only teaching is that a traversal of the tree delivers triangles in a "spatially correct" ordering. While this may be regarded as a type of "sorting", this is a sorting <u>of the leaves of the tree</u>, not of the <u>shapes at each leaf of the tree</u>, as required by the claim, and as discussed above in the arguments presented for claim 1, which are hereby incorporated by reference..

Korobkin teaches a BSP tree with single-triangle leaf nodes, precisely as described as "conventional" in the specification, and simply traverses the tree for a "spatially correct" ordering. Korobkin does not have multiple-triangle leaf nodes, and does not sort shapes <u>at each leaf of the binary-space-partition tree</u>, as required by the claim. Nor does any other cited reference teach this limitation.

It is clear, then, that no combination of Korobkin, Brokenshire, and Vlasic (or any other cited art), can meet the limitations of this claim. This claim is simply not taught or suggested by any art of record, alone or in combination, and there is no motivation to combine these references to produce a feature not found in any of them. As such, the rejection should be reversed, and this claim and all its dependent claims should be allowed over the art of record.

Claim 18

Independent Claim 18 requires:

A data processing system having at least a processor and accessible memory, comprising:

means for analyzing shapes in a graphic object;

means for creating a root node and a list of additional nodes for a binary-space-partition tree, each node associated with up to a predetermined number of at least one shape;

means for performing a partition plane selection for each additional node,

means for classifying the shapes at the additional node according to the partition plane selection; and

means for creating child nodes according to the shape classification.

This claim requires, among other limitations, means for classifying the shapes at the additional node according to the partition plane selection. The Examiner simply refers to Korobkins' teaching regarding the G-regions and U-regions in the camera view, where each region that is occupied with projected geometry is designated a G-region, and unoccupied regions are U-regions. Even if this were regarded as a "partition plane selection" for the U-region and G-region nodes, no shapes at the nodes are classified according to this partition plane selection, as claimed, and no child nodes are created according to any such shape classification, as claimed.

It is clear, then, that no combination of Korobkin, Brokenshire, and Vlasic (or any other cited art), can meet the limitations of this claim. This claim is simply not taught or suggested by any art of record, alone or in combination. As such, the rejection should be reversed, and this

claim and all its dependent claims should be allowed over the art of record.

Claim 21

Claim 21 requires:

> A computer program product tangibly embodied in a machine-readable medium, comprising:
>
> instructions for receiving node and view data for a graphic object;
>
> instructions for building a binary-space-partition tree corresponding to the graphic object, the binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf;
>
> instructions for sorting shapes at each leaf of the binary-space-partition tree; and
>
> instructions for outputting the sorted shapes.

The arguments above with respect to claim 1 apply here as well, and are incorporated by reference.

Claim 28

Independent Claim 28 requires:

> A computer program product tangibly embodied in a machine-readable medium, comprising:
>
> instructions for analyzing shapes in a graphic object;
>
> instructions for creating a root node and a list of additional nodes for a binary-space-partition tree, each node associated with up to a predetermined number of at least one shape;
>
> instructions for performing a partition plane selection for each additional node,
>
> instructions for classifying the shapes at the additional node according to the partition plane selection; and
>
> instructions for creating child nodes according to the shape classification.

The arguments above with respect to claim 8 apply here as well, and are incorporated by reference.

Ground of Rejection 2: Claims 3, 6, 7, 13, 16, 17, 23, 26, and 27 were rejected as obvious over U.S. Patent No. 6,912,293 to Korobkin ("Korobkin") in view of U.S. Patent No. 6,624,810 to Brokenshire et al. ("Brokenshire") and further in view of U.S. Patent Application Publication No. 2004/0114794 to Vlasic et al. ("Vlasic").

Claim 6

Claim 6 requires, among other limitations, that "a configuration component is used, the configuration component balancing the resolution of the binary-space-partition tree against the sorting shapes at each leaf".

As Claim 6 depends from Claim 1, the arguments above with regard to Claim 1 apply here as well, and are incorporated herein by reference, and applicant notes that the teachings missing from the proposed Korobkin/Brokenshire combination are also not taught by a Korobkin/Brokenshire/Vlasic combination. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

As no are of record teaches or suggests sorting shapes at each leaf (as opposed to sorting the leafs in the tree), no art of record can teach or suggest balancing anything against such sorting.

As can be seen, nothing in this paragraph (or any other part of Vlasic) teaches or suggests anything at all related to balancing the resolution of the binary-space-partition tree against the sorting shapes at each leaf, as required by the claim.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 16

Claim 16 requires, among other limitations, that "a configuration component is used, the configuration component balancing the resolution of the binary-space-partition tree against the sorting shapes at each leaf".

The arguments above with regard to claims 1, 6, and 11 are incorporated herein by reference.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 26

Claim 26 requires, among other limitations, that "a configuration component is used, the configuration component balancing the resolution of the binary-space-partition tree against the sorting shapes at each leaf".

The arguments above with regard to claims 1, 6, and 21 are incorporated herein by reference.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

REQUESTED RELIEF

The Board is respectfully requested to reverse the outstanding rejections and return this application to the Examiner for allowance.

Respectfully submitted,
MUNCK BUTRUS, P.C.

Date: 7/25/07

Matthew S. Anderson
Registration No. 39,093

P.O. Drawer 800889
Dallas, Texas 75380
Phone: (972) 628-3600
Fax: (972) 628-3616
E-mail: manderson@munckbutrus.com

Attorney for Applicant

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	Jianbing Huang, et al.
Serial No.:	10/731,281
Filed:	December 9, 2003
For:	SYSTEM AND METHOD FOR TRANSPARENCY RENDERING
Group No.:	2628
Examiner:	Kimbinh T. Nguyen

APPENDIX A -

Claims Appendix

1. (Previously Presented) A method for graphics processing, comprising:
   receiving node and view data for a graphic object;
   building a binary-space-partition tree corresponding to the graphic object, the binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf;
   sorting shapes at each leaf of the binary-space-partition tree; and
   outputting the sorted shapes.

2. (Original) The method of claim 1, wherein the shapes are sorted into a substantially back-to-front order.

3. (Original) The method of claim 1, further comprising caching the shape data.

4. (Original) The method of claim 1, further comprising traversing the binary-space-partition tree.

5. (Original) The method of claim 1, wherein the shapes are triangles.

6. (Original) The method of claim 1, wherein a configuration component is used, the configuration component balancing the resolution of the binary-space-partition tree against the sorting shapes at each leaf.

7. (Original) The method of claim 3, wherein a configuration component is used, the configuration component balancing resource usage against accuracy in the resolution of the caching.

8. (Previously Presented) A method for graphics processing, comprising:
analyzing shapes in a graphic object;
creating a root node and a list of additional nodes for a binary-space-partition tree, each node associated with up to a predetermined number of at least one shape;
performing a partition plane selection for each additional node,
classifying the shapes at the additional node according to the partition plane selection; and
creating child nodes according to the shape classification.

9. (Original) The method of claim 8, wherein each node represents a set of elements located in a 3-dimensional spatial region.

10. (Original) The method of claim 8, wherein the shapes are triangles.

11. (Previously Presented) A data processing system having at least a processor and accessible memory, comprising:

means for receiving node and view data for a graphic object;

means for building a binary-space-partition tree corresponding to the graphic object, the binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf;

means for sorting shapes at each leaf of the binary-space-partition tree; and

means for outputting the sorted shapes.

12. (Original) The data processing system of claim 11, wherein the shapes are sorted into a substantially back-to-front order.

13. (Original) The data processing system of claim 11, further comprising means for caching the shape data.

14. (Original) The data processing system of claim 11, further comprising means for traversing the binary-space-partition tree.

15. (Original) The data processing system of claim 11, wherein the shapes are triangles.

16. (Original) The data processing system of claim 11, wherein a configuration component is used, the configuration component balancing the resolution of the binary-space-partition tree against the sorting shapes at each leaf.

17. (Original) The data processing system of claim 13, wherein a configuration component is used, the configuration component balancing resource usage against accuracy in the resolution of the caching.

18. (Previously Presented) A data processing system having at least a processor and accessible memory, comprising:

means for analyzing shapes in a graphic object;

means for creating a root node and a list of additional nodes for a binary-space-partition tree, each node associated with up to a predetermined number of at least one shape;

means for performing a partition plane selection for each additional node,

means for classifying the shapes at the additional node according to the partition plane selection; and

means for creating child nodes according to the shape classification.

19. (Original) The data processing system of claim 18, wherein each node represents a set of elements located in a 3-dimensional spatial region.

20. (Original) The data processing system of claim 18, wherein the shapes are triangles.

21. (Previously Presented) A computer program product tangibly embodied in a machine-readable medium, comprising:

instructions for receiving node and view data for a graphic object;

instructions for building a binary-space-partition tree corresponding to the graphic object, the binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf;

instructions for sorting shapes at each leaf of the binary-space-partition tree; and

instructions for outputting the sorted shapes.

22. (Original) The computer program product of claim 21, wherein the shapes are sorted into a substantially back-to-front order.

23. (Original) The computer program product of claim 21, further comprising instructions for caching the shape data.

24. (Previously Presented) The computer program product of claim 21, further comprising instructions for traversing the binary-space-partition tree.

25. (Original) The computer program product of claim 21, wherein the shapes are triangles.

26. (Original) The computer program product of claim 21, wherein a configuration component is used, the configuration component balancing the resolution of the binary-space-partition tree against the sorting shapes at each leaf.

27. (Original) The computer program product of claim 23, wherein a configuration component is used, the configuration component balancing resource usage against accuracy in the resolution of the caching.

28. (Previously Presented) A computer program product tangibly embodied in a machine-readable medium, comprising:

instructions for analyzing shapes in a graphic object;

instructions for creating a root node and a list of additional nodes for a binary-space-partition tree, each node associated with up to a predetermined number of at least one shape;

instructions for performing a partition plane selection for each additional node,

instructions for classifying the shapes at the additional node according to the partition plane selection; and

instructions for creating child nodes according to the shape classification.

29. (Original) The computer program product of claim 28, wherein each node represents a set of elements located in a 3-dimensional spatial region.

30. (Original) The computer program product of claim 28, wherein the shapes are triangles.

DOCKET NO.: 05-03-014
U.S. Serial No. 10/731,281
PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	Jianbing Huang, et al.
Serial No.:	10/731,281
Filed:	December 9, 2003
For:	SYSTEM AND METHOD FOR TRANSPARENCY RENDERING
Group No.:	2628
Examiner:	Kimbinh T. Nguyen

APPENDIX B

Copy of Formal Drawings

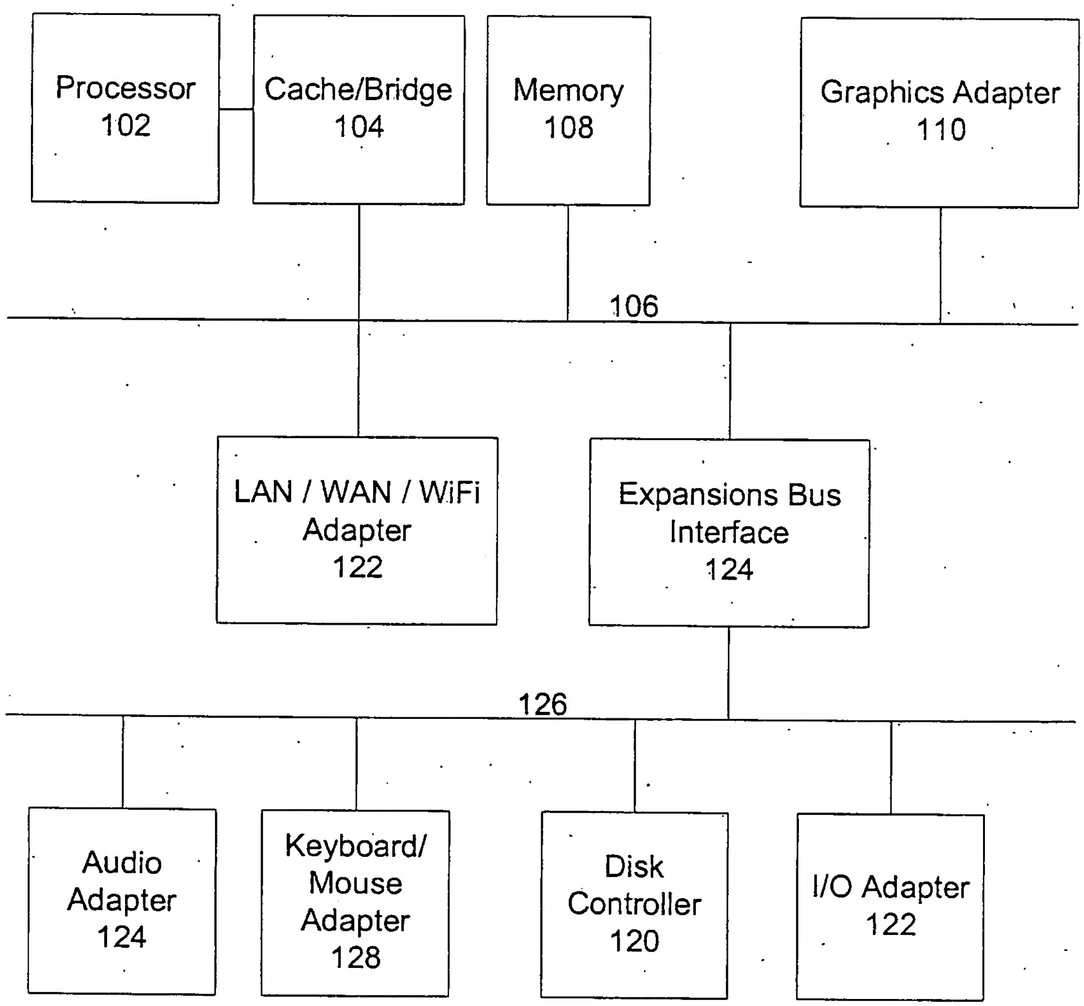
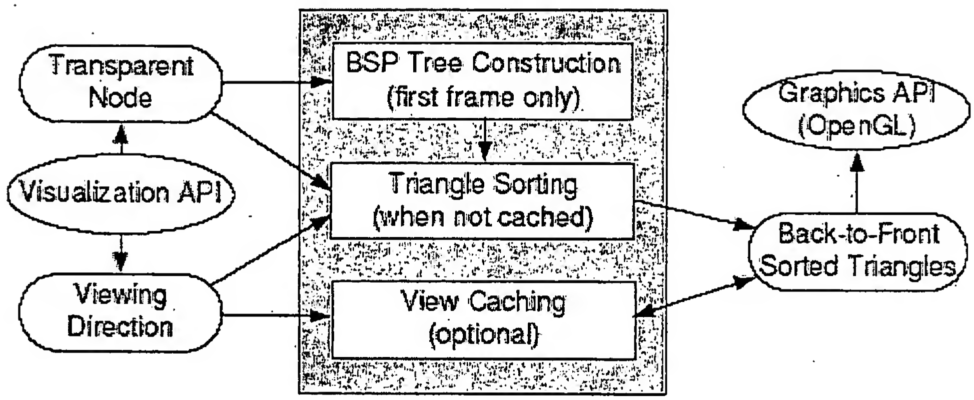
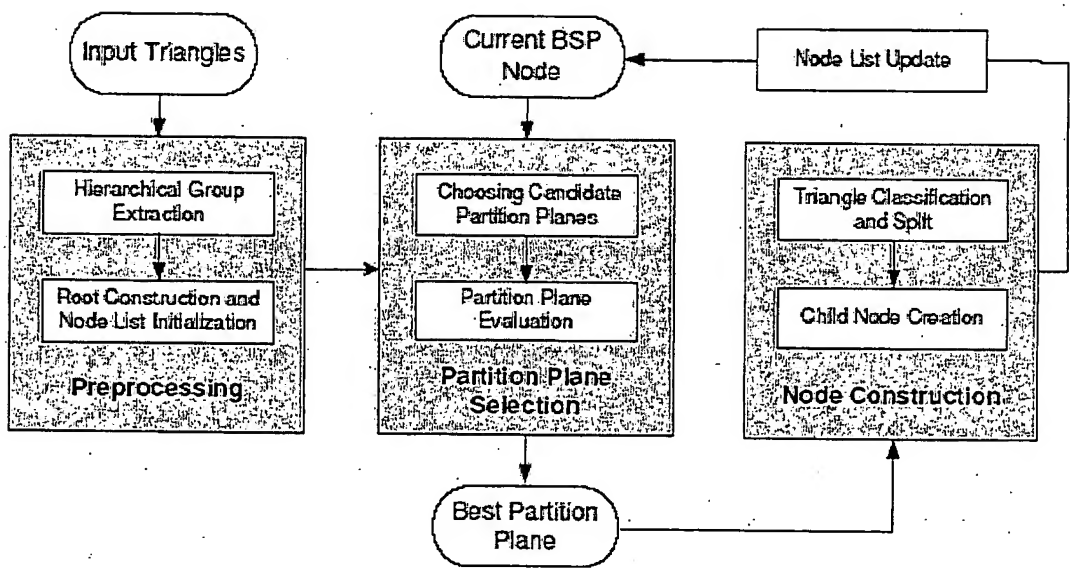

Processor
102
Cache/Bridge
104
Memory
108
Graphics Adapter
110
106
LAN / WAN / WiFi
Adapter
122
Expansions Bus
Interface
124
126
Audio
Adapter
124
Keyboard/
Mouse
Adapter
128
Disk
Controller
120
I/O Adapter
122

Figure 1



Transparent Node
Visualization API
Viewing Direction
BSP Tree Construction (first frame only)
Triangle Sorting (when not cached)
View Caching (optional)
Graphics API (OpenGL)
Back-to-Front Sorted Triangles

Figure 2



Input Triangles
Current BSP Node
Node List Update
Hierarchical Group Extraction
Root Construction and Node List Initialization
Preprocessing
Choosing Candidate Partition Planes
Partition Plane Evaluation
Partition Plane Selection
Triangle Classification and Split
Child Node Creation
Node Construction
Best Partition Plane

Figure 3

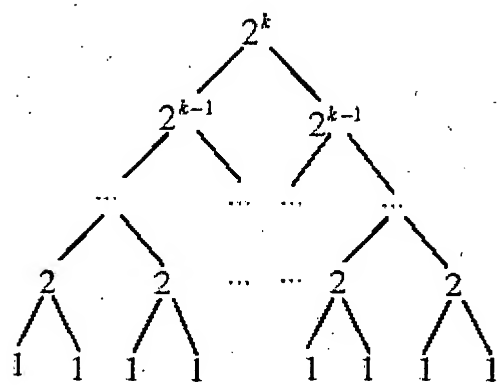
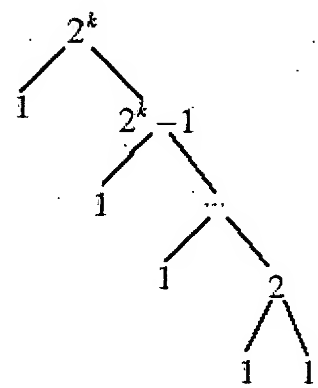


Figure 4A

Figure 4B



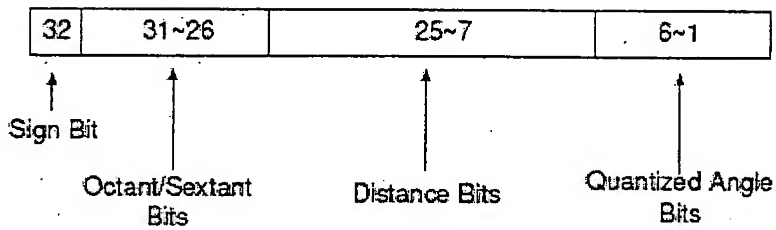


Figure 5

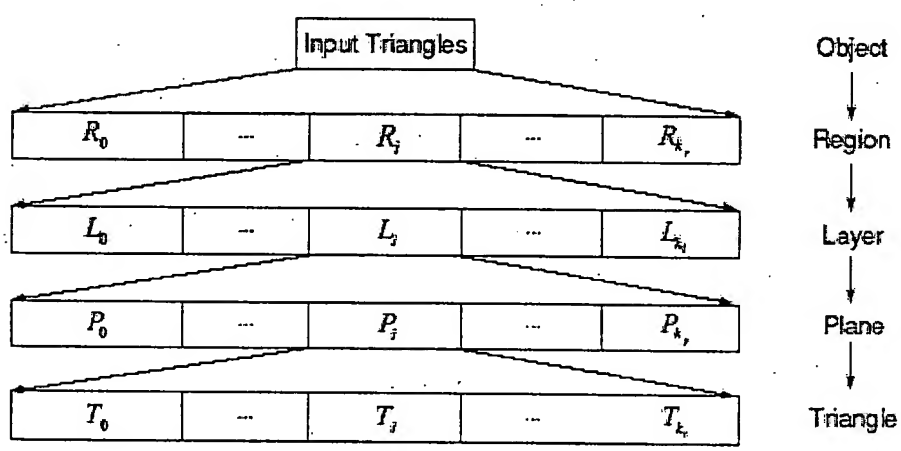


Figure 6

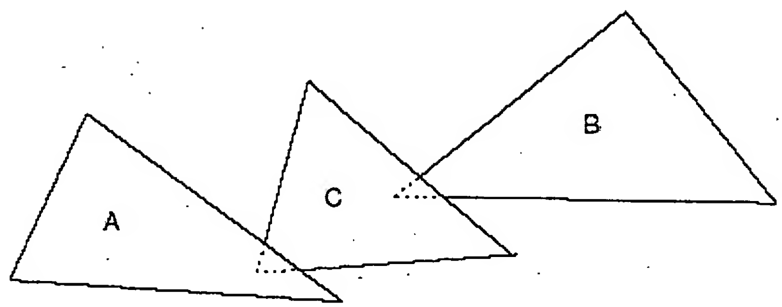


Figure 7

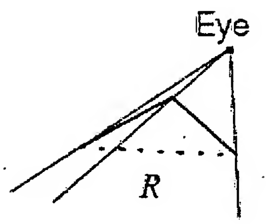


Figure 8A

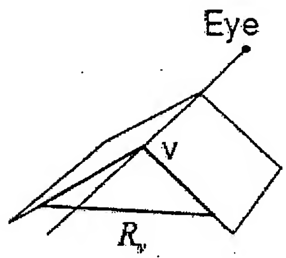


Figure 8B

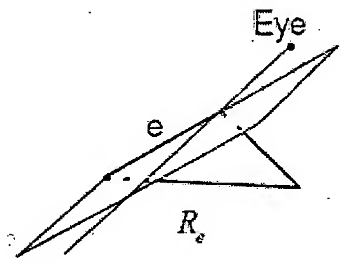


Figure 8C

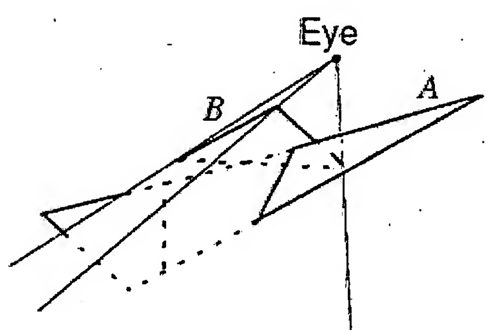


Figure 9

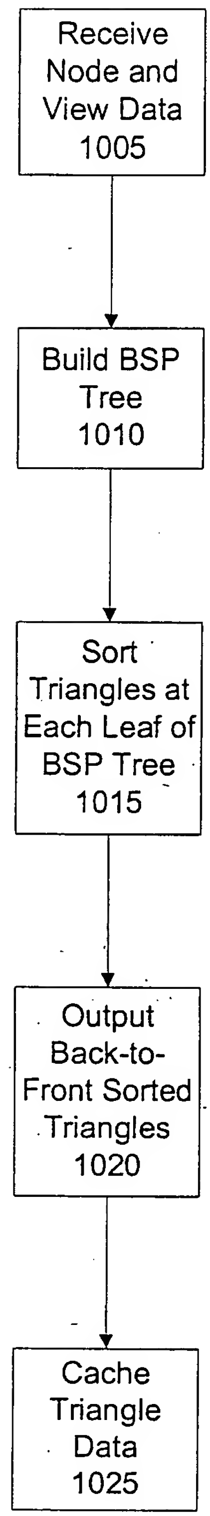

Receive Node and View Data 1005
Build BSP Tree 1010
Sort Triangles at Each Leaf of BSP Tree 1015
Output Back-to-Front Sorted Triangles 1020
Cache Triangle Data 1025

Figure 10

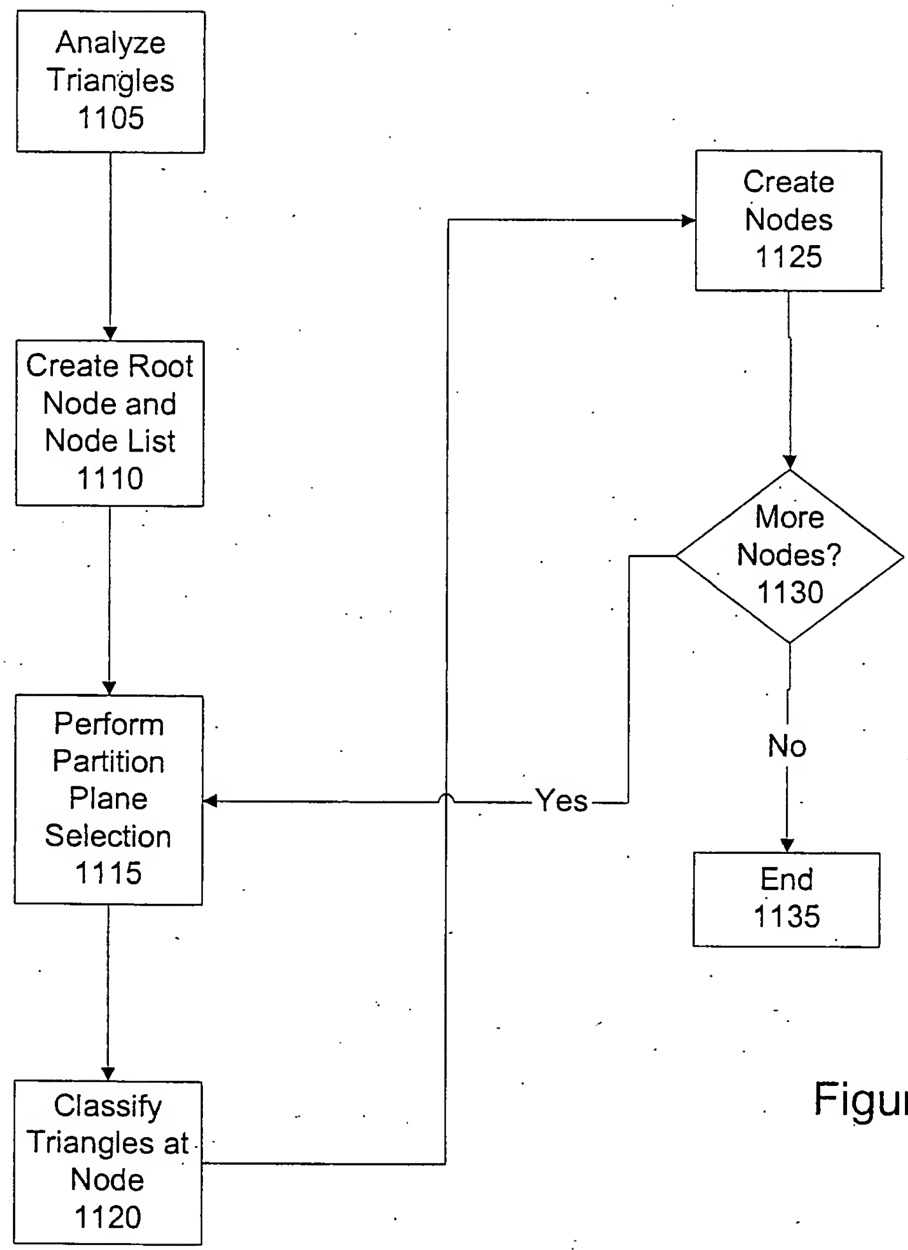

Analyze Triangles 1105
Create Root Node and Node List 1110
Perform Partition Plane Selection 1115
Classify Triangles at Node 1120
Create Nodes 1125
More Nodes? 1130
Yes
No
End 1135

Figure 11

DOCKET NO.: 05-03-014
U.S. Serial No. 10/731,281
PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	Jianbing Huang, et al.
Serial No.:	10/731,281
Filed:	December 9, 2003
For:	SYSTEM AND METHOD FOR TRANSPARENCY RENDERING
Group No.:	2628
Examiner:	Kimbinh T. Nguyen

APPENDIX C
Evidence Appendix

None.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	Jianbing Huang, et al.
Serial No.:	10/731,281
Filed:	December 9, 2003
For:	SYSTEM AND METHOD FOR TRANSPARENCY RENDERING
Group No.:	2628
Examiner:	Kimbinh T. Nguyen

APPENDIX D

Related Proceedings Appendix

None.